form, then, an internal ROM will contain all the software neded to run the system, apart from applications programs (games, word processors etc.) loaded in or written by the user. Part of this ROM will contain the code needed for converting applications programs written in BASIC into machine code (the *interpreter*); part will contain the code needed for entering and modifying user-written programs (the *editor*); and part will contain the housekeeping software needed for looking after the keyboard, displaying characters and graphics, accepting data from cassettes and allocating it to the right parts of memory and so on (the *monitor*).

The term 'monitor', not to be confused with a television or display monitor, is roughly synonymous with 'operating system'. In its simplest form, the monitor is able to do little more than accept instructions in machine code, place them in the right memory location, and supervise their execution. Once the housekeeping becomes a little more advanced than this, the monitor itself tends to be referred to as the operating system.

At the other extreme are the disk-based computers, often used in offices as small business systems, which have powerful operating systems. Before considering intermediate computers such as the Apple, we will consider the kind of operating system needed by a disk-only system.

A computer system entirely based on floppy disks for its software will normally have very little stored permanently in ROM apart from a bootstrap loader (see page 188) and a few housekeeping routines. When such a computer is switched on, the bootstrap loader contains just

enough machine code to instruct the CPU how to access a disk drive and load the operating system into main memory.

The operating system loaded into RAM has to be able to do more than the operating systems in more conventional ROM-based systems. It becomes what is called a DOS or disk operating system. An operating system such as this extends normal housekeeping functions by adding commands that act directly on the files stored on the disk. The kind of commands expected from a disk operating system include commands to list the names of the files stored on the disk, to erase or re-name files, and to copy files from the disk into main memory or onto other disks.

The operating systems of simpler, ROM-based systems usually do not have such sophisticated file-handling commands, and may have nothing more than a simple command to load a named file from tape or to store a file under a given name on a tape. A sophisticated operating system will know the exact address location on disk or tape storage where any file is stored. Less advanced operating systems may be able to do nothing more than search through all the files present until the one named is encountered, and then load it: or to write a file under a given file name on the tape at whatever point the tape happens to be when the command is issued.

Computer systems intermediate between the two are well exemplified by the BBC Micro, which is equally adept at handling cassette or disk files, using largely the same commands. The operating system resides in a ROM, but it is a physically separate ROM and can be thought of
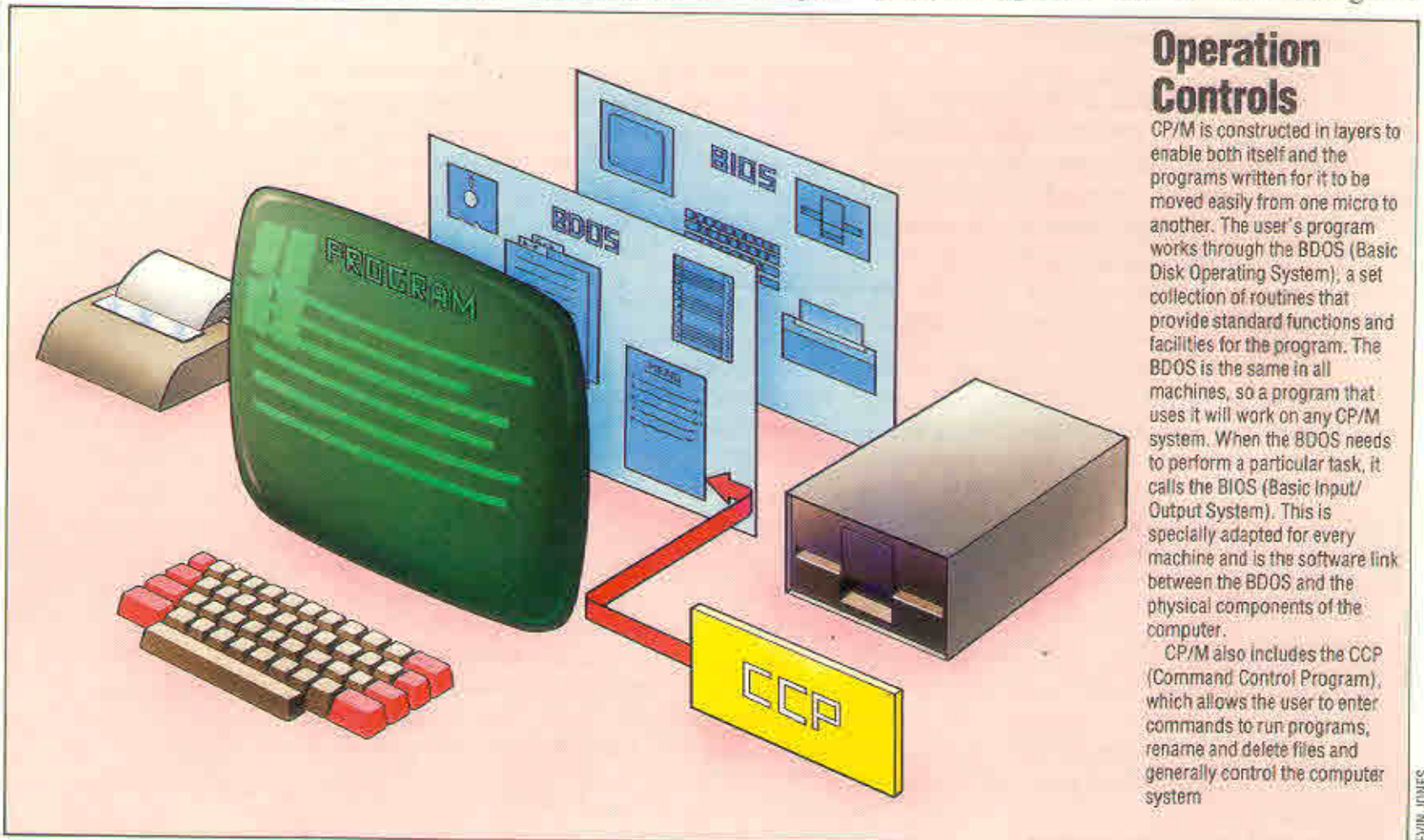
Operation Controls

CP/M is constructed in layers to enable both itself and the programs written for it to be moved easily from one micro to another. The user's program works through the BDOS (Basic Disk Operating System), a set collection of routines that provide standard functions and facilities for the program. The BDOS is the same in all machines, so a program that uses it will work on any CP/M system. When the BDOS needs to perform a particular task, it calls the BIOS (Basic Input/Output System). This is specially adapted for every machine and is the software link between the BDOS and the physical components of the computer.

CP/M also includes the CCP (Command Control Program), which allows the user to enter commands to run programs, rename and delete files and generally control the computer system